US-PAT-NO: 6100875

DOCUMENT-IDENTIFIER: US 6100875 A

TITLE: Keyboard pointing device

---------- KWIC ---------

In one embodiment, each data packet comprises three bytes of binary data and
has several fields. For example, some fields of the data packet contain data
indicating the state (inactive or active) of the left button key or right
button key, fields contain a number of X and/or Y counts (i.e., increments of
directional movement in the X and/or Y directions, as utilized by the mouse
driver to move the pointer on the screen) and fields contain data indicating
whether an overflow has occurred in the X counts or Y counts.

Typically, one bit in each data packet represents the state of the left button.
In the present embodiment, the state of this bit stays the same for consecutive
packets in accordance with the last change of this bit. Therefore, at the
decision block 506, the keyboard controller 202 performs a test to determine if
the left button key 120 has changed state since the last test. If the left
button key 120 has not changed state since the last test, then this indicates
that the user was holding the left button key 120 down during the last pass
through this procedure, and the bit in the data packet reflecting the state of
the left button key 120 need not be changed in the next packet sent to the host
302. Thus, control passes to a decision block 510. However, if the left
button key 120 has changed state since the last test, this indicates that the

user has just activated the left button key 120 and the bit in the next data packet which reflects the state of the left button should be set. When the packet is sent to the host 302, the mouse driver interprets the bit to determine that the left button of the mouse was just depressed. Thus, control passes to a process block 508, and data indicating that the left mouse button switch state has changed to closed is placed into a data packet (i.e., the left button state bit is set in the present embodiment). Additionally, the keyboard controller 202 sets a "packet changed" flag (not part of the mouse data packet) which indicates to the keyboard controller 202 that the data packet has changed. Thereafter, control passes to the decision block 510.

Once any information indicating that the left button key 120 has changed state is included within the data packet, the keyboard controller 202 determines if the right button key 122 (i.e., the key assigned to represent the right button on a mouse) is currently active, as represented in the decision block 510. If the right button key 122 is active, then control passes to a decision block 512, and a determination is made whether the state of the right button key 122 has changed since the last test. If the right button key 122 has not changed state since the last test, then this indicates that the user was holding the right button key 122 down during the last execution of the subroutine, and a bit in the packet representing the state of the right mouse button already represents the current state. Thus, control passes to a decision block 516 (FIG. 5b). However, if the right button key 122 has changed state since the last test, this indicates that the user has just activated the right button key 122, and the bit in the data packet, which represents the

state of the right
button key 122 should be set. This will indicate to the host (via the mouse
driver) that the right button of the mouse was just activated. Therefore,
control passes to a process block 514, and the keyboard controller 202 sets the
bit in the current data packet indicating that the state of the right button
has changed to closed (i.e., a bit is set in the data packet currently being
constructed for transmission to the host 302). In addition, the keyboard
controller 202 sets the packet changed flag. Thereafter, control passes to the
decision block 516 (FIG. 5b).